CLARE: A Joint Approach to Label Classification and Tag Recommendation

Yilin Wang[1], Suhang Wang[1], Jiliang Tang[2], Guojun Qi[3], Huan Liu[1], Baoxin Li[1]
[1]Arizona State University
[2]Michigan State University
[3]University of Central Florida
{yilinwang,suhang.wang,huan.liu,baoxin.li}@asu.edu,tangjili@msu.edu,guojun.qi@ucf.edu

Abstract

Data classification and tag recommendation are both important and challenging tasks in social media. These two tasks are often considered independently and most efforts have been made to tackle them separately. However, labels in data classification and tags in tag recommendation are inherently related. For example, a Youtube video annotated with NCAA, stadium, pac12 is likely to be labeled as football, while a video/image with the class label of coast is likely to be tagged with beach, sea, water and sand. The existence of relations between labels and tags motivates us to jointly perform classification and tag recommendation for social media data in this paper. In particular, we provide a principled way to capture the relations between labels and tags, and propose a novel framework CLARE, which fuses data CLAssification and tag REcommendation into a coherent model. With experiments on three social media datasets, we demonstrate that the proposed framework CLARE achieves superior performance on both tasks compared to the state-of-the-art methods.

Introduction

The increasing popularity of social media generates massive data at an unprecedented rate. For example, on average for every minute, 300 hours of video are uploaded to YouTube [1], 54.9 million posts are shared on Reddit[2] and 30 billion photos are posted on Instagram [3]. Therefore many techniques (or tasks) have been proposed to help organize and access social media data, among which classification and tag recommendation are two popular ones. For social media posts[4], classification is to assign them class labels (Kaplan and Haenlein 2010; Agichtein *et al.* 2008; Li and Zaiane 2015), while tag recommendation aims to suggest tags to annotate them automatically (Wang *et al.* 2016; Sigurbjörnsson and Van Zwol 2008).

For each task of classification and tag recommendation, we have witnessed a large body of literature in recent years (Chen *et al.* 2013; Sigurbjörnsson and Van Zwol 2008; Wang *et al.* 2015a; Zhang *et al.* 2014; Wang *et al.* 2015b; Lian *et al.* 2015). For social media posts, labels in classification often capture their high-level content, while tags in tag recommendation are likely to describe their attributes such as objects in images, actions in videos and keywords in blogs. Therefore classification and tag recommendation are generally considered as two independent tasks and the majority of efforts are made to study them separately. However, these two tasks should be connected since labels and tags of social media posts are often related(Wang *et al.* 2009). For example, posts annotated with NCAA, stadium, pac12 are likely to be labeled as football, while posts with the class label of president campaign are likely to be tagged with election, polling, democratic and republic. In other words, tags could provide evidence for class labels, which could in turn serve as useful contextual information for tags. However, in (Wang *et al.* 2009), there is no explicit relations between tags and labels, which results a relative poor performance on each task. Aforementioned intuitions motivate us to develop a more robust joint classification and tag recommendation framework for social media posts.

In this paper, we investigate the problem of predicting class labels and tags simultaneously for social media posts by exploiting the relations between labels and tags. The differences between traditional methods and the proposed method are illustrated in Figure 1. As shown in Figure 1(a), traditional methods treat these two problems separately – classification uses data and its labels to learn a classifier (or a label predictor), while tag recommendation uses data and its tags to learn a tag predictor. In contrast, the proposed framework performs classification and tag recommendation jointly by leveraging data, labels, tags and relations between labels and tags as demonstrated in Figure 1(b). Since the current methods cannot take advantage of relations between labels and tags, we proceed to study two fundamental problems: (1) how to capture relations between labels and tags mathematically; and (2) how to make use of it for joint classification and tag recommendation. These two problems are tackled by the proposed framework CLARE and our contributions are summarized as follows:

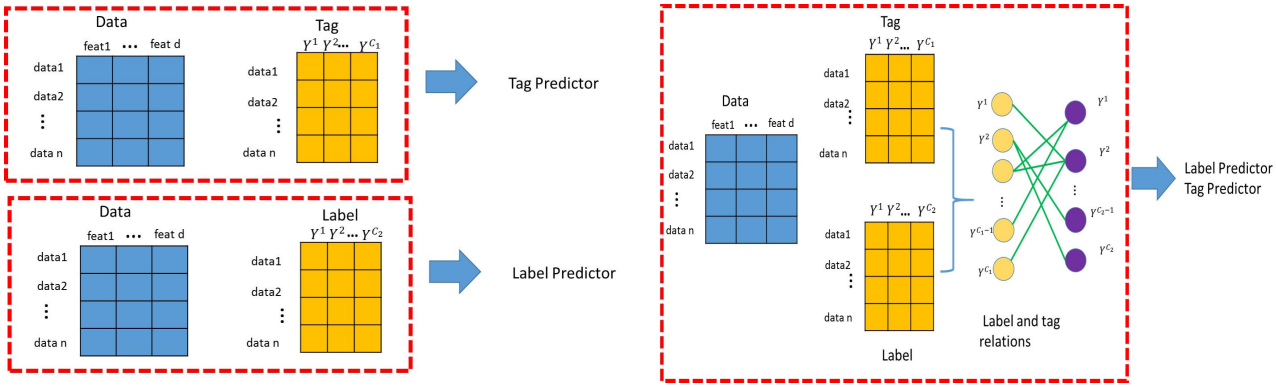
- We provide a principled approach to model relations between labels and tags, which bridges the tasks of classification and recommendation;

[1] $https://www.youtube.com/yt/press/statistics.html$

[2] $http://www.redditblog.com/2014_12_01_archive.html$

[3] $https://instagram.com/press/$

[4] In this paper, we use posts in a loose sense to cover blogs, microblogs, images and videos.



(a) Traditional Methods (b) The Proposed Method

Figure 1: Differences between existing methods and the proposed method.

- We propose a novel joint framework CLARE, which can predict class labels and tags for social media posts simultaneously and achieve better performance than the current state-of-the-art methods on both tasks. It is worth noting that the proposed algorithm explores the joint relations between labels and tags, however, it do NOT require a test example come with labels or tags to annotate one another;
- We conduct experiments on various social media datasets to analyze and understand the inter-working of the proposed framework CLARE.

The framework CLARE

Before detailing the proposed framework, we first introduce notations we used in this paper. We use $\mathbf{X} \in \mathbb{R}^{n \times d}$ to denote a set of social media posts where n is the number of posts and d is the number of features. Note that there are various ways to extract features to represent social media posts such as raw features and features learned via deep learning techniques (Jia *et al.* 2014). Let $\mathbf{Y}_t \in \mathbb{R}^{n \times c_1}$ be the post-tag matrix where c_1 is the number of tags. $\mathbf{Y}_t(i,j) = 1$ if the i-th post is annotated the j-th tag and $\mathbf{Y}_t(i,j) = 0$ otherwise. Similarly we use $\mathbf{Y}_c \in \mathbb{R}^{n \times c_2}$ to represent the class label affiliation matrix where c_2 is the number of class labels. $\mathbf{Y}_c(i,j) = 1$ if the i-th post is labeled as the j-th class and $\mathbf{Y}_c(i,j) = 0$ otherwise. In the following subsections, we first introduce basic models for classification and tag recommendation, then detail the model component to capture relations between labels and tags and finally discuss the proposed framework.

Basic Models

For classification, we assume that there is a linear classifier $\mathbf{W}_c \in \mathbb{R}^{d \times c_2}$ to map $\mathbf{X}$ to $\mathbf{Y}_c$ as $\mathbf{Y}_c = \mathbf{X}\mathbf{W}_c$. $\mathbf{W}_c$ can be obtained by solving the following optimization problem:

$$\min_{\mathbf{W}_c} \ \Omega(\mathbf{W}_c) + \mathcal{L}(\mathbf{X}\mathbf{W}_c, \mathbf{Y}_c) \tag{1}$$

where $\mathcal{L}()$ is a loss function and Ω is a regularization penalty to avoid overfitting. Popular choices of $\mathcal{L}$ include square, logistic and hinge loss functions.

For tag recommendation, we also assume that there is a linear function $\mathbf{W}_t \in \mathbb{R}^{d \times c_1}$ which captures the relation between $\mathbf{X}$ and $\mathbf{Y}_t$ as $\mathbf{Y}_t = \mathbf{X}\mathbf{W}_t$. Similarly the optimization problem to learn $\mathbf{W}_t$ is:

$$\min_{\mathbf{W}_t} \ \Omega(\mathbf{W}_t) + \mathcal{L}(\mathbf{X}\mathbf{W}_t, \mathbf{Y}_t) \tag{2}$$

Combining Eq. (1) and Eq. (2), we can obtain a unified basic model for classification and tag recommendation as:

$$\min_{\mathbf{W}} \ \Omega(\mathbf{W}) + \mathcal{L}(\mathbf{X}\mathbf{W}, \mathbf{Y}) \tag{3}$$

where $\mathbf{Y} = [\mathbf{Y}_t, \mathbf{Y}_c] \in \mathbb{R}^{n \times (c_1 + c_2)}$ and $\mathbf{W} = [\mathbf{W}_t, \mathbf{W}_c] \in \mathbb{R}^{d \times (c_1 + c_2)}$. Note that though we rewrite the basic models of classification and tag recommendation into a unified formulation, we still consider them as two independent tasks since we do not capture any relations between these two tasks.

Capturing Relations between Labels and Tags

In the previous subsection, we defined a unified formulation for classification and tag recommendation. Capturing relations between labels and tags can further pave a way for us to develop a joint framework that enables simultaneous classification and tag recommendation.

The relations between labels and tags can be denoted as a bipartite graph as shown in Figure 1(b). We assume that $\mathbf{B} \in \mathbb{R}^{c_2 \times c_1}$ is the adjacency matrix of the graph where $\mathbf{B}(i,j) = 1$ if both the i-th label and the j-th tag co-occur in the same posts and $\mathbf{B}(i,j) = 0$ otherwise. Note that in this paper, we do not consider the concurrence frequencies of tags and labels and we would like to leave it as one future work. From the bipartite graph, we can identify groups of labels and tags that share similar properties such as semantical meanings. A feature $\mathbf{X}(:,j)$ should be either relevant or irrelevant to labels and tags in the same group. In $\mathbf{W}$, $\mathbf{W}_c(i,j)$ indicates the effect of the i-th feature on predicting the j-th label; while $\mathbf{W}_t(i,k)$ denotes the impact of the i-th feature on the k-th tag. Therefore we can impose constraints on $\mathbf{W}$, which are derived from group information on the bipartite graph, to capture relations between labels and tags.

In this paper, we use overlapped group lasso to extract groups from the bipartite graph – for the i-th label, we consider the label and tags that connect to that label in the bipartite graph as a group, i.e., $\mathbf{B}(i,j)=1$. Note that a tag may connect to several labels thus groups formed via the aforementioned process have **overlaps**. Assume that $\mathcal{G}$ is the set of groups we detect from the label-tag bipartite graph and we propose to minimize the following term to capture relations between labels and tags as:

$$\sum_{i=1}^{d}\sum_{g\in\mathcal{G}}\alpha_g\left\|\mathbf{w}_g^i\right\|_2 \tag{4}$$

where α_g is the confidence of the group g and $\mathbf{w}_g^i$ is a vector concatenating $\{\mathbf{W}(i,j)\}_{j\in g}$. For example, if $g=\{1,5,9\}$, $\mathbf{w}_g^i=[\mathbf{W}(i,1),\mathbf{W}(i,5),\mathbf{W}(i,9)]$.

Next we discuss the inner workings of Eq. (4). Let us check the terms in Eq. (4) related to a specific group g, $\sum_{i=1}^{d}\left\|\mathbf{w}_g^i\right\|_2$, which is equal to adding a ℓ_1 norm on the vector $\mathbf{g}=[\mathbf{w}_g^1,\mathbf{w}_g^2,\ldots,\mathbf{w}_g^d]$, i.e., $\|\mathbf{g}\|_1$. That ensures a sparse solution of $\mathbf{g}$; in other words, some elements of $\mathbf{g}$ could be zeros. If $\mathbf{g}_i=0$ or $\|\mathbf{w}_g^i\|_2=0$, the effects of the i-th feature on both the label and tags in the group g are eliminated simultaneously.

The Proposed Framework

Incorporating the component to capture relations between labels and tags leads us to the following joint framework for classification and tag recommendation:

$$\min_{\mathbf{W}}\ \Omega(\mathbf{W})+\mathcal{L}(\mathbf{XW},\mathbf{Y})+\alpha\sum_{i=1}^{d}\sum_{g\in\mathcal{G}}\alpha_g\left\|\mathbf{w}_g^i\right\|_2 \tag{5}$$

where the first and second terms are the basic models for classification and tag recommendation; and the third term captures the relations of these two tasks. The parameter α controls the contribution of the third term.

In this paper, we choose square loss as the loss function $\mathcal{L}$ and the ridge regularization as the regularization penalty Ω. With these choices, the optimization problem for the proposed framework CLARE can be rewritten as:

$$\min_{\mathbf{W}}\quad \|\mathbf{XW}-\mathbf{Y}\|_F^2+\alpha\sum_{i=1}^{d}\sum_{g\in\mathcal{G}}\alpha_g\left\|\mathbf{w}_g^i\right\|_2+\beta\|\mathbf{W}\|_F^2 \tag{6}$$

An Optimization Algorithm for CLARE

Since the group structures are overlapped, directly optimizing the objective function is difficult. We propose to use alternating directions method of multipliers (ADMM)(Boyd *et al.* 2011; Yogatama and Smith 2014) to optimize it. The central idea in ADMM is to break the optimization problem down into subproblems, each depending on a subset of the dimensions of $\mathbf{W}$. Specially, we introduce an auxiliary variable $\mathbf{V}\in\mathbb{R}^{d\times c_2(c_1+c_2)}$, and for each subproblem i, we encode the group constrains to $\mathbf{V_i}$ which has the same dimension of the group size. Therefore the objective function to be minimized by ADMM is:

$$\begin{aligned}\min_{\mathbf{W},\mathbf{V}}\ &\|\mathbf{XW}-\mathbf{Y}\|_F^2+\beta\|\mathbf{W}\|_F^2+\\ &\sum_{i=1}^{d}\sum_{j=1}^{c_2}\alpha_j\left\|\mathbf{V}(i,(c_1+c_2)\cdot(j-1)+1:(c_1+c_2)\cdot j)\right\|_2\\ &s.t.\mathbf{V}=\mathbf{WM}\end{aligned} \tag{7}$$

where α_j means the confidence of the j-th group and $\mathbf{M}\in\{0,1\}^{(c_1+c_2)\times c_2(c_1+c_2)}$ is defined as: if the i-th tag connects to the j-th label, then $M(i,(c_1+c_2)(j-1)+i)=1$, otherwise it is zero. Here we do not assume any prior knowledge available on the group weight. In the following, for simplicity and without loss of generality, we assume $\alpha_j=\alpha,\forall j$.

For brevity, we denote $\mathcal{L}(\mathbf{X},\mathbf{Y},\mathbf{W})=\|\mathbf{XW}-\mathbf{Y}\|_F^2$, $\Omega_{group}(\mathbf{V})$ to be the group regularizer, and Ω_{reg} to be the F-norm regularizer. To apply ADMM, we use augmented Lagrangian of Equation 7:

$$\begin{aligned}&\Omega_{group}(\mathbf{V})+\Omega_{reg}(\mathbf{W})+\mathcal{L}(\mathbf{X},\mathbf{Y},\mathbf{W})+\\ &\qquad Tr(\mu^T(\mathbf{V}-\mathbf{WM}))+\frac{\rho}{2}\|\mathbf{V}-\mathbf{WM}\|_F^2\end{aligned} \tag{8}$$

where μ is the Lagrange variables, and $\rho>0$ is the parameter that controls the quadratic penalty.

Updating W

If $\mathbf{V}$ and μ are fixed, the objective function is decoupled and the constraints are independent of $\mathbf{W}$. Thus we can optimize $\mathbf{W}$ separately and ignore the terms without $\mathbf{W}$, leading to the following:

$$\begin{aligned}&\min_{\mathbf{W}}\ \mathcal{J}(\mathbf{W})=\Omega_{reg}(\mathbf{W})-Tr(\mu^T\mathbf{WM})+\mathcal{L}(\mathbf{X},\mathbf{Y},\mathbf{W})+\\ &\frac{\rho}{2}\|\mathbf{V}-\mathbf{WM}\|_F^2\\ &\cong\min_{\mathbf{W}}\quad\Omega_{reg}(\mathbf{W})+\mathcal{L}(\mathbf{X},\mathbf{Y},\mathbf{W})+\frac{\rho}{2}\left\|\mathbf{WM}-(\mathbf{V}+\frac{\mu}{\rho})\right\|_F^2\\ &=Tr((\mathbf{XW}-\mathbf{Y})^T(\mathbf{XW}-\mathbf{Y}))+\beta Tr(\mathbf{W}^T\mathbf{W})+\\ &\frac{\rho}{2}Tr([\mathbf{WM}-(\mathbf{V}+\frac{\mu}{\rho})]^T[\mathbf{WM}-(\mathbf{V}+\frac{\mu}{\rho})])\end{aligned} \tag{9}$$

Taking the derivation of $\mathcal{J}(\mathbf{W})$ and setting it to zero, we obtain the following form:

$$\begin{aligned}&(\frac{1}{2}\beta\mathbf{I}+\mathbf{X}^T\mathbf{X})\mathbf{W}-\mathbf{X}^T\mathbf{Y}+\mathbf{W}(\frac{\rho}{2}\mathbf{MM}^T+\frac{1}{2}\beta\mathbf{I})\\ &-\frac{\rho}{2}(\mathbf{V}+\frac{\mu}{\rho})\mathbf{M}^T=0\end{aligned} \tag{10}$$

In Equation 10, solving $\mathbf{W}$ is intractable. On the other hand, $\mathbf{X}^T\mathbf{X}+\frac{1}{2}\beta\mathbf{I}$ and $\frac{\rho}{2}\mathbf{MM}^T+\frac{1}{2}\beta\mathbf{I}$ are symmetric and positive definite. Thus we employ eigen decomposition for each of them:

$$\begin{aligned}\frac{1}{2}\beta\mathbf{I}+\mathbf{X}^T\mathbf{X}&=\mathbf{U}_1\mathbf{\Lambda}_1\mathbf{U}_1^T\\ \frac{\rho}{2}\mathbf{MM}^T+\frac{1}{2}\beta\mathbf{I}&=\mathbf{U}_2\mathbf{\Lambda}_2\mathbf{U}_2^T\end{aligned} \tag{11}$$

where $\mathbf{U}_1$, $\mathbf{U}_2$ are eigen vectors and $\mathbf{\Lambda}_1$, $\mathbf{\Lambda}_2$ are diagonal matrices with eigen values on the diagonal. Substituting $\mathbf{X}^T\mathbf{X} + \frac{1}{2}\beta\mathbf{I}$ and $\frac{\rho}{2}\mathbf{M}\mathbf{M}^T + \frac{1}{2}\beta\mathbf{I}$ in Eq 10:

$$\mathbf{U}_1\mathbf{\Lambda}_1\mathbf{U}_1^T\mathbf{W} - \mathbf{X}^T\mathbf{Y} + \mathbf{W}\mathbf{U}_2\mathbf{\Lambda}_2\mathbf{U}_2^T - \frac{\rho}{2}(\mathbf{V} + \frac{\mu}{\rho})\mathbf{M}^T = 0 \tag{12}$$

Multiplying $\mathbf{U}_1^T$ and $\mathbf{U}_2$ from left to right on both sides, and letting $\widetilde{\mathbf{W}} = \mathbf{U}_1^T\mathbf{W}\mathbf{U}_2$ and $\mathbf{Q} = \mathbf{U}_1^T[\mathbf{X}^T\mathbf{Y} + \frac{\rho}{2}(\mathbf{V} + \frac{\mu}{\rho})\mathbf{M}^T]\mathbf{U}_2$, Eq 12 becomes:

$$\mathbf{\Lambda}_1\widetilde{\mathbf{W}} + \widetilde{\mathbf{W}}\mathbf{\Lambda}_2 = \mathbf{Q} \tag{13}$$

Then, we can obtain $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ as:

$$\widetilde{\mathbf{W}}(s,l) = \frac{\mathbf{Q}(s,l)}{\lambda_1^s + \lambda_2^l} \tag{14}$$

$$\mathbf{W} = \mathbf{U}_1\widetilde{\mathbf{W}}\mathbf{U}_2^T \tag{15}$$

where λ_1^s is the s-th eigen value of $\mathbf{X}^\mathbf{T}\mathbf{X} + \frac{1}{2}\beta\mathbf{I}$ and λ_2^l is l-th eigen value of $\frac{\rho}{2}\mathbf{M}\mathbf{M}^\mathbf{T} + \frac{1}{2}\beta\mathbf{I}$.

Updating for $\mathbf{V}$

If $\mathbf{W}$ and μ are fixed, the $\mathbf{V}$ can be obtained by the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{V}} \quad \Omega_{group}(\mathbf{V}) + Tr(\mu^T\mathbf{V}) + \frac{\rho}{2}\left\|\mathbf{V} - \mathbf{W}\mathbf{M}\right\|_F^2 \\ \cong &\min_{\mathbf{V}} \quad \Omega_{group}(\mathbf{V}) + \frac{\rho}{2}\left\|\mathbf{V} - (\mathbf{W}\mathbf{M} - \frac{\mu}{\rho})\right\|_F^2 + const \end{aligned} \tag{16}$$

When applied to the collection of groups for the parameters, $\mathbf{V}, \Omega_{group}(\mathbf{V})$ no longer have overlapping groups. We denote the j-th group in i-th row as $\mathbf{V}_{i,j} = \mathbf{V}(i, (c_1 + c_2) \cdot (j-1) + 1 : (c_1 + c_2) \cdot j)$, similarly for $\mathbf{M}_{i,j}$, which is defined as the columns of M corresponding to the groups. Hence we can solve the problem separately for each row of $\mathbf{V}$ within one group:

$$\min_{\mathbf{V}_{i,j}} \alpha\left\|\mathbf{V}_{i,j}\right\|_2 + \frac{\rho}{2}\left\|\mathbf{V}_{i,j} - ((\mathbf{W}\mathbf{M})_{i,j} - \frac{\mu_{i,j}}{\rho})\right\|_2^F \tag{17}$$

Note that Eq 17 is the proximal operator (Yuan *et al.* 2011) of $\frac{1}{\rho}(\mathbf{V})_{i,j}$ applied to $(\mathbf{W}\mathbf{M})_{i,j} - \frac{\mu_{i,j}}{\rho}$. Let $Z_{i,j} = (\mathbf{W}\mathbf{M})_{i,j} - \frac{\mu_{i,j}}{\rho}$. The solution by applying the proximal operator used in non-overlapping group lasso to each subvector is:

$$\begin{aligned} \mathbf{V}_{i,j} &= prox_{\Omega_{group},}(\mathbf{Z}_{i,j}) \\ &= \begin{cases} \mathbf{0} & if\ \left\|\mathbf{Z}_{i,j}\right\|_2 \leqslant \frac{\alpha}{\rho} \\ \frac{\left\|\mathbf{Z}_{i,j}\right\|_2 - \frac{\alpha}{\rho}}{\left\|\mathbf{Z}_{i,j}\right\|_2}\mathbf{Z}_{i,j} & otherwise. \end{cases} \end{aligned} \tag{18}$$

Updating for μ

Updating μ is simple and is defined as $\mu = \mu + \rho(\mathbf{V} - \mathbf{W}\mathbf{M})$.

With the updating rules, the proposed algorithm for CLARE is summarized in Algorithm 1.

Algorithm 1 An Optimization Algorithm for CLARE

Input: $\{\mathbf{X}, \mathbf{Y}, \mathbf{M}\}\alpha, \beta, \mu$
Output: c_1 tags label and c_2 data labels for each data instance.
Initialization: $\mathbf{W} = 0$
Precompute Eigen vectors $\mathbf{U_1}, \mathbf{U_2}$ eigen values $\mathbf{\Lambda_1}, \mathbf{\Lambda_2}$
while Not Converge **do**
  Calculate $\mathbf{Q} = \mathbf{U}_1^T[\mathbf{X}^T\mathbf{Y} + \frac{\rho}{2}(\mathbf{V} + \frac{\mu}{\rho})\mathbf{M}^T]\mathbf{U}_2$
  Compute $\widetilde{\mathbf{W}}$ use Eq 14 and update $\mathbf{W} = \mathbf{U}_1\widetilde{\mathbf{W}}\mathbf{U}_2^T$
  parfor $i \leftarrow 1, d$ **do** (computed in parallel)
    for $j \leftarrow 1, c_2$ **do**
      $\mathbf{V}_{i,j} = prox_{\Omega_{group},}(\mathbf{Z}_{i,j})$
    end for
  end parfor
  Update $\mu = \mu + \rho(\mathbf{V} - \mathbf{W}\mathbf{M})$
end whileEnd
Using max-pooling for $\mathbf{X}\mathbf{W}$ to predict tags and labels.

Convergence Analysis

Since the sub-problems are convex for $\mathbf{W}$ and $\mathbf{V}$, respectively, Algorithm 1 is guaranteed to converge because they satisfy the two assumptions required by ADMM. The proof of the convergence can be found in (Boyd *et al.* 2011). Specially, Algorithm 1 has dual variable convergence. Our empirical results show that our algorithm often converges within 100 iterations for all the datasets we used for evaluation.

Time Complexity Analysis

The main computation cost for $\mathbf{W}$ involves the eigen decomposition on $\mathbf{X}^\mathbf{T}\mathbf{X} + \frac{1}{2}\beta\mathbf{I}$ and $\frac{1}{2}\rho\mathbf{M}\mathbf{M}^\mathbf{T} + \frac{1}{2}\beta\mathbf{I}$; and the computation of $\mathbf{Q} = \mathbf{U}_1^T[\mathbf{X}^T\mathbf{Y} + \frac{\rho}{2}(\mathbf{V} + \frac{\mu}{\rho})\mathbf{M}^T]\mathbf{U}_2$. The time complexity for Eigen decomposition is $O(d^3)$. However, in Algorithm 1 the Eigen decomposition is only computed once before the loop. The computation cost for $\mathbf{Q}$ is $O(nd^2)$ due to the sparsity of $\mathbf{M}$. The computation of $\mathbf{V}$ depends on the proximal method within each group. Since there are c_2 groups which have the group size $c_1 + c_2$ for each feature dimension, the total computation cost for $\mathbf{V}$ is $O(dc_2(c_1 + c_2))$. It is worth noting that $\mathbf{V}$ can be computed in parallel for each feature dimension. Similarly, the computational cost of μ depends on the computation of $\mathbf{W}\mathbf{M}$, which is $O(dc_1^2)$.

Experimental Analysis

In this section, we conduct experiments to evaluate the effectiveness of CLARE. After introducing datasets and experimental settings, we compare CLARE with the state-of-the art methods of tag recommendation and classification. Further experiments are conducted to investigate the effects of important parameters in CLARE.

Experiments Settings

The experiments are conducted on 3 publicly available social media datasets.

USAA-YouTube dataset (Fu *et al.* 2014): All 1500 videos in the dataset are unstructured and unedited which are either taken by digital camera or mobile phones and they are originally uploaded on the social media website Youtube. It has 8 class labels with totally 69 manually annotated tags. These eight classes are birthday party, graduation party, music performance, non-music performance, parade, wedding ceremony, wedding dance and wedding reception. The size of tag vocabulary is 69 and example tags are: dancing, eating, fast-moving, garden, living room, conversation, presentation, bride, flag, and candles. We use three types of features including: scale invariant feature transform (SIFT), Mel-frequency cepstral coefficients (MFCC), and spatial temporal interest points (STIP).

NUS-Flickr dataset (Chua *et al.* 2009): It contains 269,648 images crawled from Flickr. The images are linked to 5,000 different user tags, which are annotated by users registered on Flickr. Beyond these images and user tags, the images are labeled with 81 concepts. We use the most common 1k tags in our experiment. We also filter out those images with less than 7 tags, resulting in 110k images. 500-dimensional visual features are extracted using a bag-of visual-word model with local SIFT descriptor.

Shoe-Zappo dataset (Yu and Grauman 2014): It is a large shoe dataset consisting of 50,025 catalog images collected from Zappos.com. The images are divided into 4 major categories: shoes, sandals, slippers, and boots. The tags are functional types and individual brands such as high-heel, oxford, leather, lace up, and pointed toe. The number of tags is 147 and we extract LAB color features and GIST features (Oliva and Torralba) as (Yu and Grauman 2014).

We use accuracy as the metric to assess classification performance since all datasets are relatively balanced. To evaluate the performance of tag recommendation, we rank all tags based on their relevant scores and return the top K ranked tags. We use the average precision $AP@K$ as the evaluation metric which has been widely used in the literature (Chen *et al.* 2013; Lin *et al.* 2013; Wang *et al.* 2009). Although, the proposed model can incorporate any advanced features, e.g., CNN feature, we use the features provided by (Fu *et al.* 2014),(Chua *et al.* 2009), and (Yu and Grauman 2014), respectively, for fair comparison. [5].

Performance Comparison

We compare CLARE with the following representative algorithms:

- SVM (Chang and Lin 2011): It uses the state of the art classifier SVM for classification with linear kernel; We also apply it to tag prediction by considering tags as a kind of labels;
- GLasso (Yuan and Lin 2006): The original framework is to handle high-dimensional and multi-class data. To extend it for joint classification and tag recommendation, we also consider tags as a kind of labels and apply GLasso to learn the mapping of features to labels and tags. Note that it does not make use of the label-tag bipartite graph. We use the implementation in (Liu *et al.* 2009);
- sLDA (Wang *et al.* 2009): It is a joint framework based on topic models, which learns both class labels and annotations given latent topics;
- LS (Ji *et al.* 2008): A multi label classification method that exploits the label correlation information. To apply LS for joint classification and tag recommendation, we consider tags as a kind of labels and use tag and label relations to replace the label correlation in the original model; and
- FT (Chen *et al.* 2013): It is one of the state-of-the art annotation method which is based on linear mapping and co-regularized joint optimization. To apply it for classification, we consider labels as tags to annotate; and
- RD: It predicts labels and tags by randomly guessing.

For all baseline methods with parameters, we use cross-validation to determine their values. Each time we choose 60% of a dataset as training data and the remaining as testing. Since only sLDA and GLasso are the baselines for joint label prediction, we present the result separately for demonstration.

From the tables, we make the following observations:

- The proposed method considers classification and tag recommendation jointly in a bipartite graph structure tends to outperform the methods which treat them separately. These results support that (1) tags can provide evidence for the classification; especially for the NUS-Flickr dataset that contains 81 label classes, those methods utilize information from tags significantly improve the classification performance. (2) The performance of tag recommendation $AP@K$ indicates that the class label contains important information for tag recommendation;
- The proposed method with model components to capture relations between labels and tags outperform those without. For example, compared to GLasso and sLDA, the proposed framework, modeling the label-tag bipartite graph, gain remarkable performance improvement for both classification and tag recommendation; and
- Most of the time, the proposed framework CLARE performs the best among all the baselines, which demonstrates the effectiveness of the proposed algorithm. There are two major reasons. First, CLARE jointly performs classification and recommendation. Second, CLARE captures relations between labels and tags by extracting group information from the label-tag bipartite group, which works as the bridge between classification and tag recommendation. More details about the effect of the relations between labels and tags on CLARE will be discussed in the following subsection.

Parameter Analysis

There are two important parameters for the proposed framework CLARE – α controlling the contribution from the model component of capturing relations between labels and

[5]If we use advanced features, it would be hard to tell whether the performance gain comes from the feature or the model we proposed.

Table 1: Performance comparison in terms of classification.

Method	USAA-YouTube(8 class)	NUS-Flickr (81 class)	Shoe-Zappo (4 class)
SVM	36.15%	18.92%	75.57 %
GLasso	38.25%	27.51%	76.31%
sLDA	32.28%	26.12%	74.32%
LS	39.39%	34.39%	86.03%
FT	39.26%	35.67%	85.39%
RD	12.49%	1.23 %	25.01%
CLARE	**53.07%**	**40.32%**	**89.39%**

Table 2: Performance comparison in terms of tag recommendation.

Method	USAA-YouTube (69 tags)			NUS-Flikr (1k tags)			Shoe-Zappo (147 tags)		
	AP@3	AP@5	AP@10	AP@3	AP@5	AP@10	AP@3	AP@5	AP@10
SVM	58.45%	53.53%	48.95%	18.71%	13.12%	10.92%	52.29%	46.17%	38.07%
GLasso	59.32%	55.12%	47.31%	17.50%	13.43%	10.11%	58.32%	49.22%	42.31%
sLDA	37.32%	31.12%	17.31%	18.95%	14.26%	11.78%	61.32%	57.12%	49.31%
LS	61.96%	57.77 %	50.94%	**22.69%** %	**17.21%**	13.35%	73.56%	66.42%	61.49%
FT	62.42%	57.52%	51.78%	21.35%	16.77%	13.43%	69.01%	60.77%	57.85%
RD	1.44%	1.43%	1.44%	0.10%	0.11%	0.11%	0.67%	0.67%	0.68%
CLARE	**77.10%**	**71.08%**	**62.95%**	21.22%	16.18%	**13.94%**	**76.74%**	**69.47%**	**63.71%**

tags and β controlling the regularization penalty. In this subsection, we investigate the impact of these parameters on the performance of the proposed framework.

To study the impact of α, we fix $\beta = 0.1$ and vary the value of α as $\{10^{-6}, 0.01, 0.1, 0.2, 0.3, 0.5, 0.7, 1.5, 2, 10, 100\}$. The performance variance w.r.t α is shown in Figure 2. Due to the page limitation, we only show results from Shoe-Zappo since we have similar observations on other datasets. In general, with the increase of α, the CLARE performance increases significantly first, reaches its peak and then drops with larger values. Especially, when α increases from 10^{-6} to 0.01, the CLARE performance increases almost 8%, which suggests the importance of the model component to capture relations between labels and tags. When α is in $[0.3, 1]$, the performance is relatively stable. When α increases from 10 to 100, the performance decreases dramatically. Indicating the model component will dominate the learning process and the learned parameters could overfit.

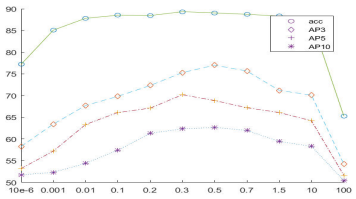


Figure 2: Performance variance w.r.t. α. Note that the Y axis is the performance and X axis is the value of α.

To study the impact of β, we vary the values of β as $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 10, 10^{2}, 10^{4}, 10^{6}, \}$ and fix $\alpha = 0.3$. The performance of CLARE with the changes of β is demonstrated in Figure 3. When β is small, CLAER may overfit, which leads to poor performance. When β becomes larger, the learned $\mathbf{W}$ will have larger amount of shrinkage and values in $\mathbf{W}$ become more robust to collinearity. However, if the β becomes extremely large, the regularization penalty dominates the learning process and elements of $\mathbf{W}$ tend to be zeros, which results in poor performance as well.

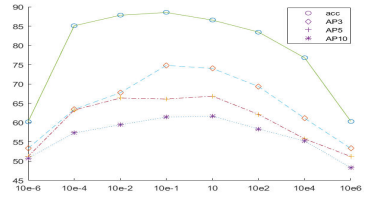


Figure 3: Performance variance w.r.t. β. Note that the Y axis denotes the performance and X axis is the value of β.

Conclusion and Future Work

Due to the relations between labels and tags, we study the problem of joint classification and tag recommendation in this paper. We extract group information from the label-tag bipartite graph as constraints to bridge classification and tag recommendation and propose a novel framework CLARE that performs classification and tag recommendation simultaneously. Experiments on three social media datasets demonstrate that: (1) joint classification and recommendation can improve performance for each task; and (2) the importance to consider relations between tags and labels. In the future, we will include robust community detection to extract the group in order to avoid noise tags.

Acknowledgment

Yilin Wang and Baoxin Li were supported in part by an ARO grant (#W911NF1410371) and an ONR grant (#N00014-15-1-2722). G.-J. Qi is partly sponsored by NSF Grant 1560302.Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of ARO, ONR or NSF.

References

Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. Finding high-quality content in social media. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 183–194. ACM, 2008.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

Minmin Chen, Alice Zheng, and Kilian Weinberger. Fast image tagging. In *Proceedings of the 30th international conference on Machine Learning*, pages 1274–1282, 2013.

Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, page 48. ACM, 2009.

Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shaogang Gong. Learning multimodal latent attributes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(2):303–316, 2014.

Shuiwang Ji, Lei Tang, Shipeng Yu, and Jieping Ye. Extracting shared subspace for multi-label classification. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 381–389. ACM, 2008.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*, pages 675–678. ACM, 2014.

Andreas M Kaplan and Michael Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53(1):59–68, 2010.

Jundong Li and Osmar Zaiane. Associative classification with statistically significant positive and negative rules. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 2015.

Wenzhao Lian, Piyush Rai, Esther Salazar, and Lawrence Carin. Integrating features and similarities: Flexible models for heterogeneous multiview data. 2015.

Zijia Lin, Guiguang Ding, Mingqing Hu, Jianmin Wang, and Xiaojun Ye. Image tag completion via image-specific and tag-specific linear sparse reconstructions. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013.

Jun Liu, Shuiwang Ji, Jieping Ye, et al. Slep: Sparse learning with efficient projections. *Arizona State University*, 6:491, 2009.

Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*.

Börkur Sigurbjörnsson and Roelof Van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW*, pages 327–336. ACM, 2008.

Chong Wang, David Blei, and Fei-Fei Li. Simultaneous image classification and annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1903–1910. IEEE, 2009.

Suhang Wang, Jiliang Tang, Yilin Wang, and Huan Liu. Exploring implicit hierarchical structures for recommender systems. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 1813–1819. AAAI Press, 2015.

Yilin Wang, Suhang Wang, Jiliang Tang, Huan Liu, and Baoxin Li. Unsupervised sentiment analysis for social media images. In *International Joint Conferences on Artificial Intelligence*, 2015.

Yilin Wang, Suhang Wang, Jiliang Tang, Huan Liu, and Baoxin Li. PPP: Joint pointwise and pairwise image label prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Dani Yogatama and Noah Smith. Making the most of bag of words: Sentence regularization with alternating direction method of multipliers. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 656–664, 2014.

A. Yu and K. Grauman. Fine-Grained Visual Comparisons with Local Learning. In *Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Lei Yuan, Jun Liu, and Jieping Ye. Efficient methods for overlapping group lasso. In *Advances in Neural Information Processing Systems*, pages 352–360, 2011.

Qiang Zhang, Jiayu Zhou, Yilin Wang, Jieping Ye, and Baoxin Li. Image cosegmentation via multi-task learning. In *BMVC*, 2014.